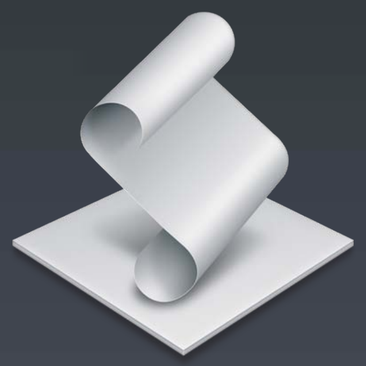
# Writing AppleScript Automator Actions

## Session 130

These are confidential sessions—please refrain from filming or taking pictures.

**Application Technologies**

# Writing AppleScript Automator Actions



**Todd Fernandez**
AppleScript and Automator Engineering Manager

http://developer.apple.com/wwdc2005/

# Agenda

- Introduction
- Hands-on Example Project
- More Information
- Q&A

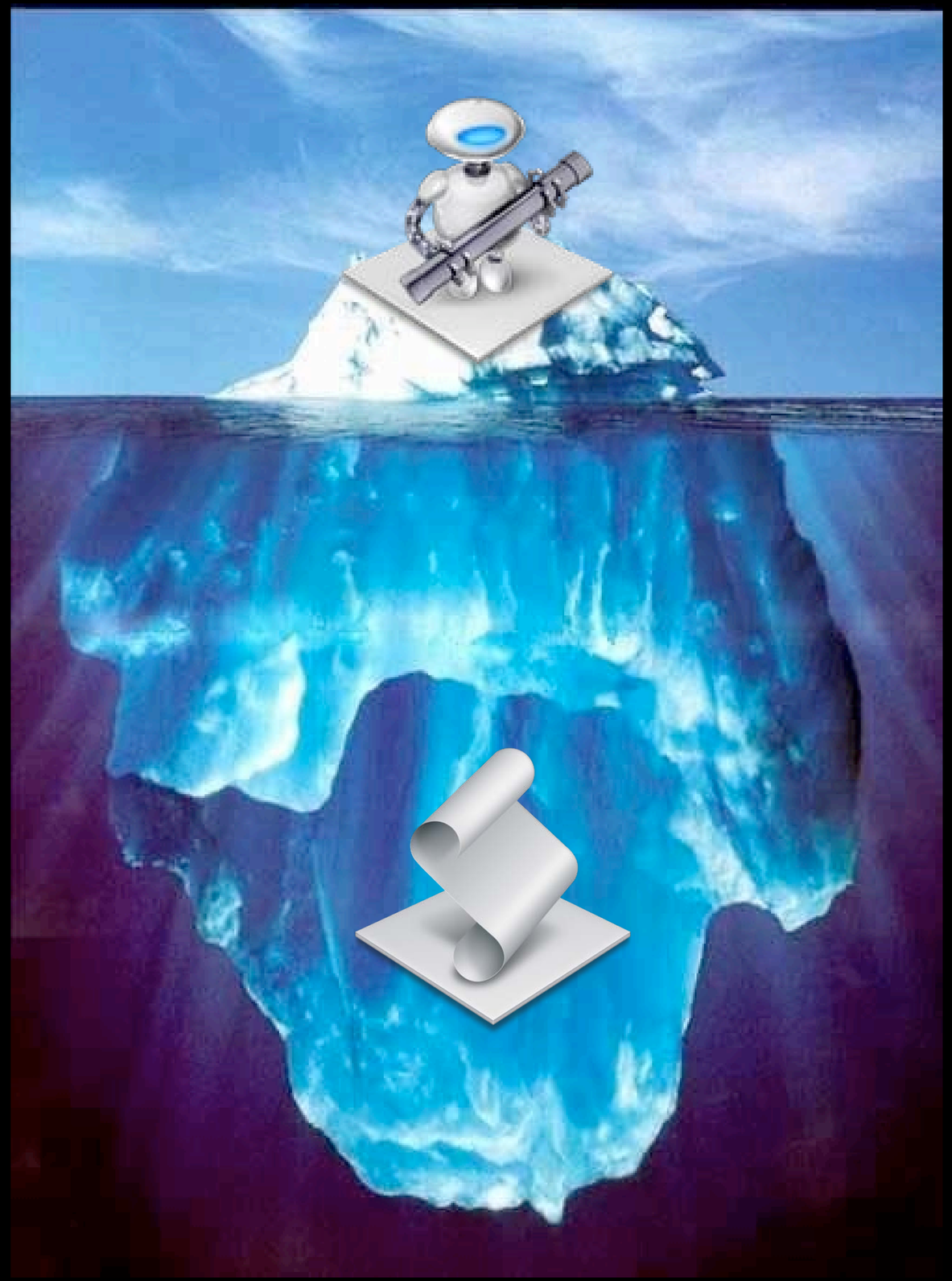
# Introduction

http://developer.apple.com/wwdc2005/

“Automator is a fantastic boon to Mac users. Not just the geeks: it's a gift to ordinary users…. Half the ginchiness of Automator is how well-integrated it is into the rest of the OS.”

**Andy Ihnatko, the Mac Observer**

# Thanks!



# Automator

http://developer.apple.com/wwdc2005/

# What Is Automator?

## Personal Automation Assistant

- Application to create workflows
- Large set of actions
- Tools to create actions
- Examples
  - Actions
  - Workflows

# Automator Terminology

## Action

- Created by developers and scripters
- Discrete bit of functionality
- Example: Copy Finder Items

# Automator Terminology

## Action

- Where do users get them?
  - Included with Automator for Apple applications
  - Third-party application developers
  - Some users will create their own

# Automator Terminology

## Workflow

- Created by users
- Composed of actions
- Example: email to the members of my Address Book group “family” a new archive of a copy of every image with keyword “family” cropped 15% and sepia filtered

# Automator Terminology

## Workflow

- Created by users
- Composed of actions
- Example: email to the members of my Address Book group “family” a new archive of a copy of every image with keyword “family” cropped 15% and sepia filtered

# Automator Terminology

## Workflow

- Created by users
- Composed of actions
- Example: email to the members of my Address Book group “family” a new archive of a copy of every image with keyword “family” cropped 15% and sepia filtered

# What You'll Learn

- How to build an AppleScript Automator action
  - Create and manage the project using Xcode
  - Build the user interface using Interface Builder
  - Manage the user interface using
    - Cocoa Bindings
    - AppleScript Studio
  - Implement using AppleScript
  - Provide key information in the Info.plist

# Hands-on Format

- Session materials
  - Prebuilt project for each step
    - Require Xcode 2.1
  - Detailed outline with all script source
- Options
  - Build action with me
  - Follow along in current milestone or completed project
  - Just watch me now and experiment later

# Creating an Action

http://developer.apple.com/wwdc2005/

# Creating an Action

## Types and requirements

- Can be created using
  - AppleScript
  - Cocoa
  - Scripting languages, e.g., bash, Perl, Python
- Targeted application must have
  - A scripting interface (AppleScript)
  - APIs (Cocoa)
  - Or be a command line tool (Scripts)

| | |
|---|---|
| Hands On: Writing Cocoa Automator Actions | Session 137<br>Thursday 3:30PM |
| Building Automator Actions for System Administrators | Session 645<br>Friday 5:00PM |

# Creating an Action

## Technologies and tools we'll use today

- AppleScript
- AppleScript Studio
- Cocoa Bindings
- Interface Builder
- Xcode
  - Action target inspector (new in Xcode 2.1)

| | |
|---|---|
| Hands On: AppleScript Studio in Depth | Session 406<br>Friday 9:00AM |
| Getting the Most Out of Cocoa Bindings | Session 136<br>Thursday 2:00PM |

# Duplicate Finder Items Step 0

**The Final Product**

http://developer.apple.com/wwdc2005/

# Hands On:

## Duplicate Finder Items Step 0

**The Final Product**

http://developer.apple.com/wwdc2005/

# Duplicate Finder Items Step 1

**Simple Initial Version**

http://developer.apple.com/wwdc2005/

# What You Will Do in Step 1

- Create a new AppleScript Automator Action project
- Write a simple script to implement the action's behavior
- Build and test the action
- Tour the action's default behavior

# Project Layout

## Step 1

- main.applescript
- Info.plist
- English.lproj
  - InfoPlist.strings
  - main.nib

# Hands On:
## Duplicate Finder Items Step 1

**Simple Initial Version**

http://developer.apple.com/wwdc2005/

# What You Learned in Step 1

- How to create a new AppleScript Automator Action project
- How to implement the action using AppleScript
- How to build and test the action

# Duplicate Finder Items Step 2

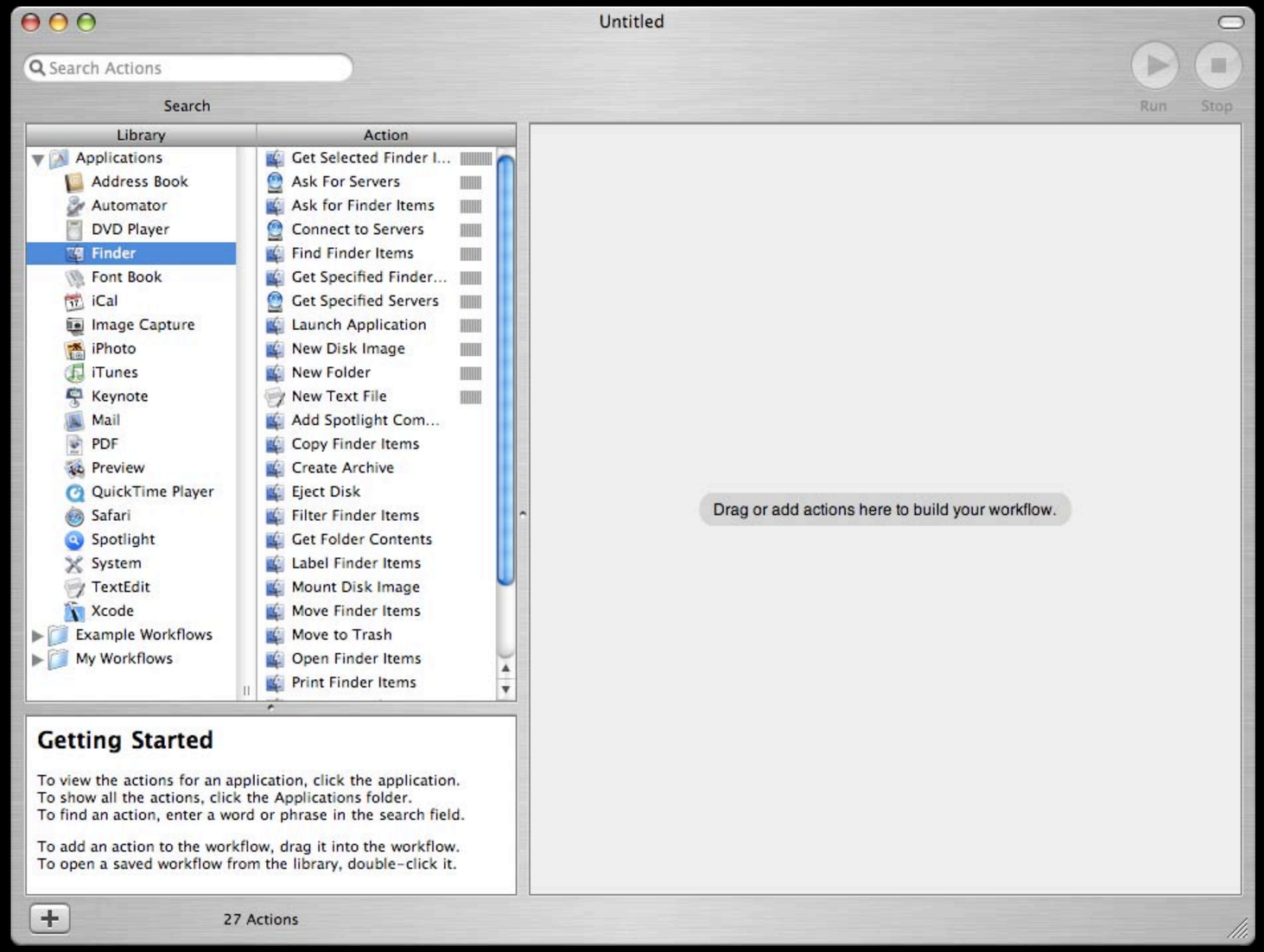
**Configure Info.plist**

http://developer.apple.com/wwdc2005/

# Info.plist Keys

## General

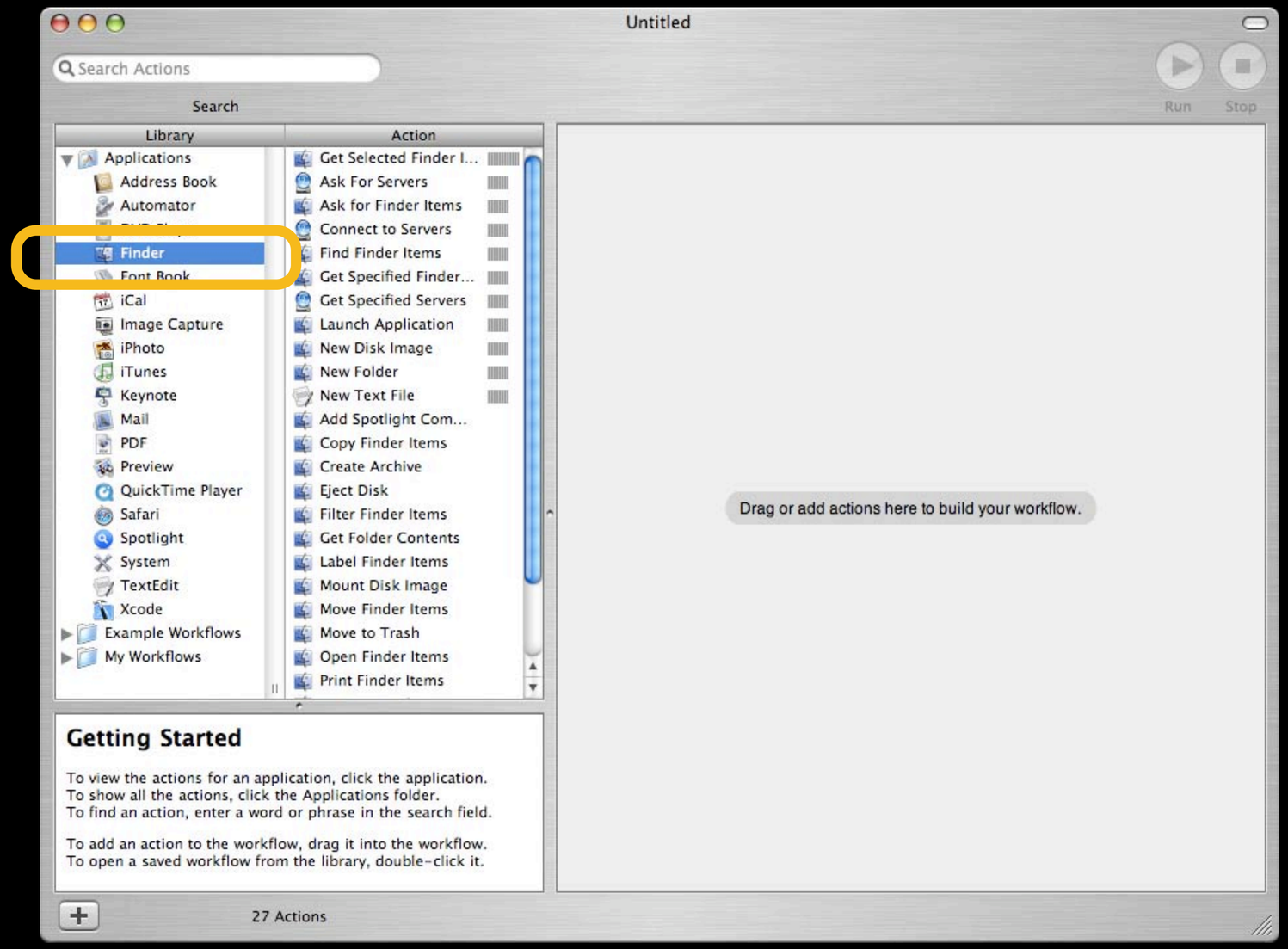
- AMApplication
- AMCategory
  - Use one of ours
  - Create your own
- AMIconName
- AMName



# Info.plist Keys

## General

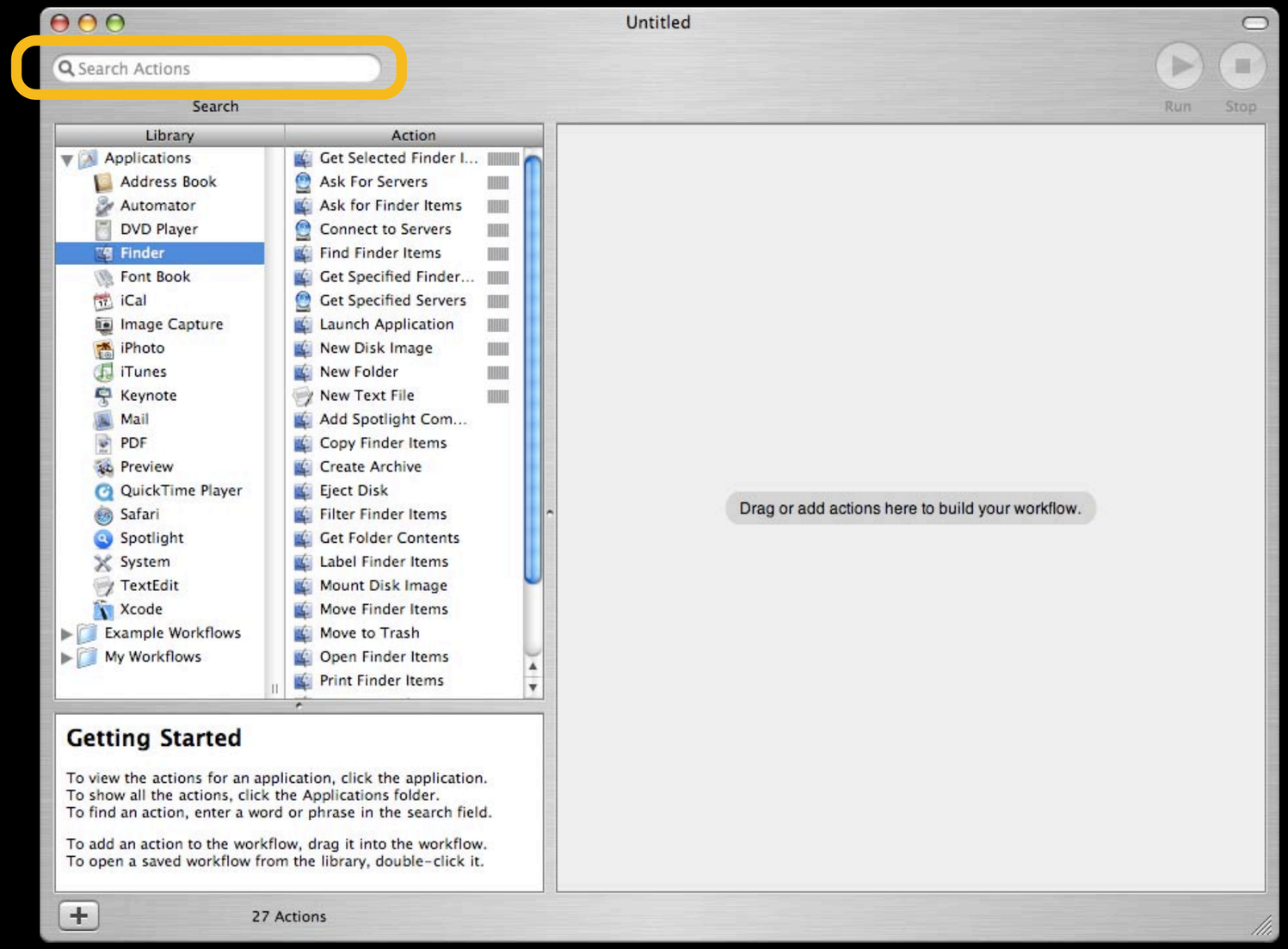
- AMApplication
- AMCategory
  - Use one of ours
  - Create your own
- AMIconName
- AMName



# Info.plist Keys

## General

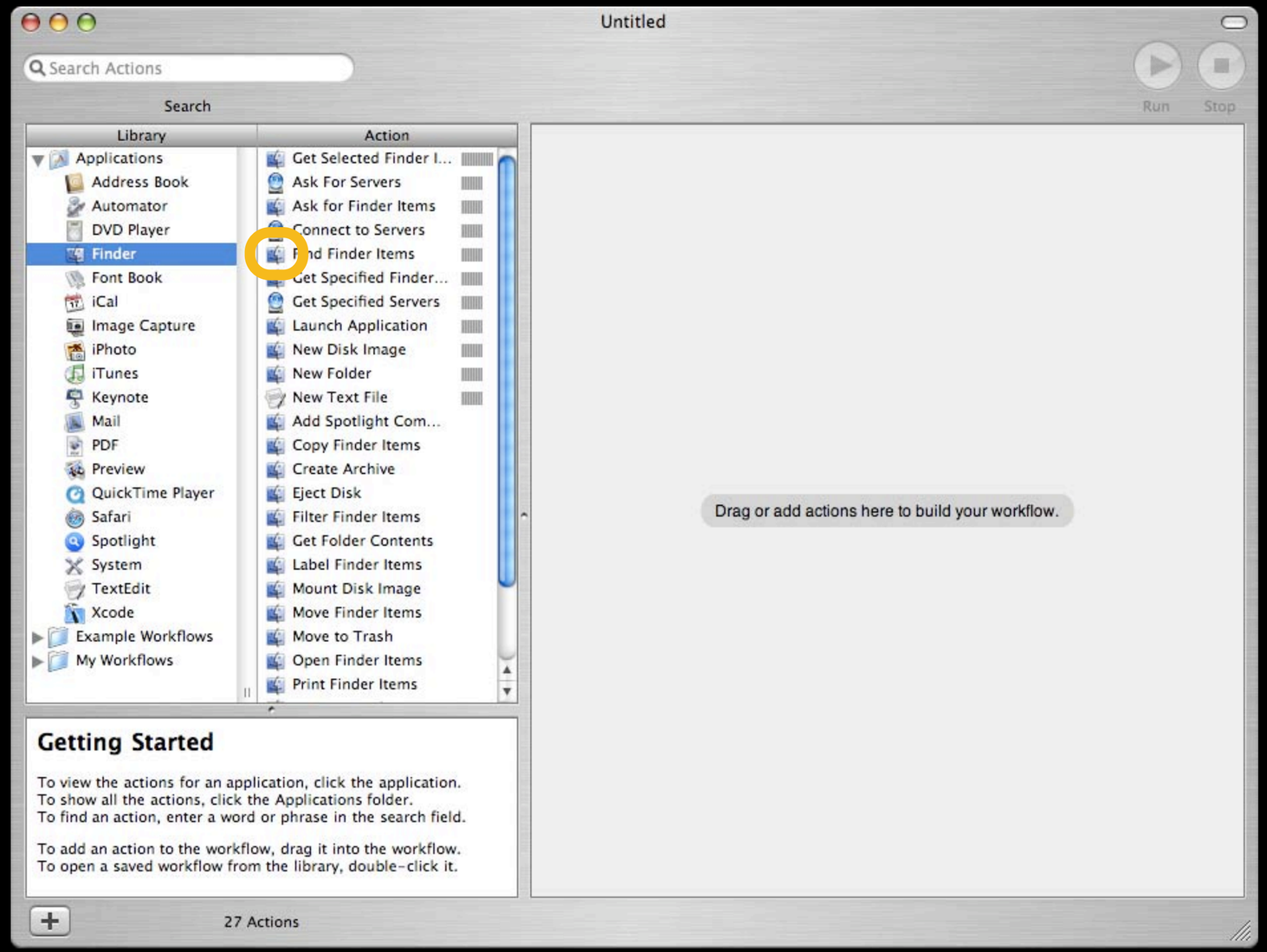
- AMApplication
- AMCategory
  - Use one of ours
  - Create your own
- AMIconName
- AMName



# Info.plist Keys

## General

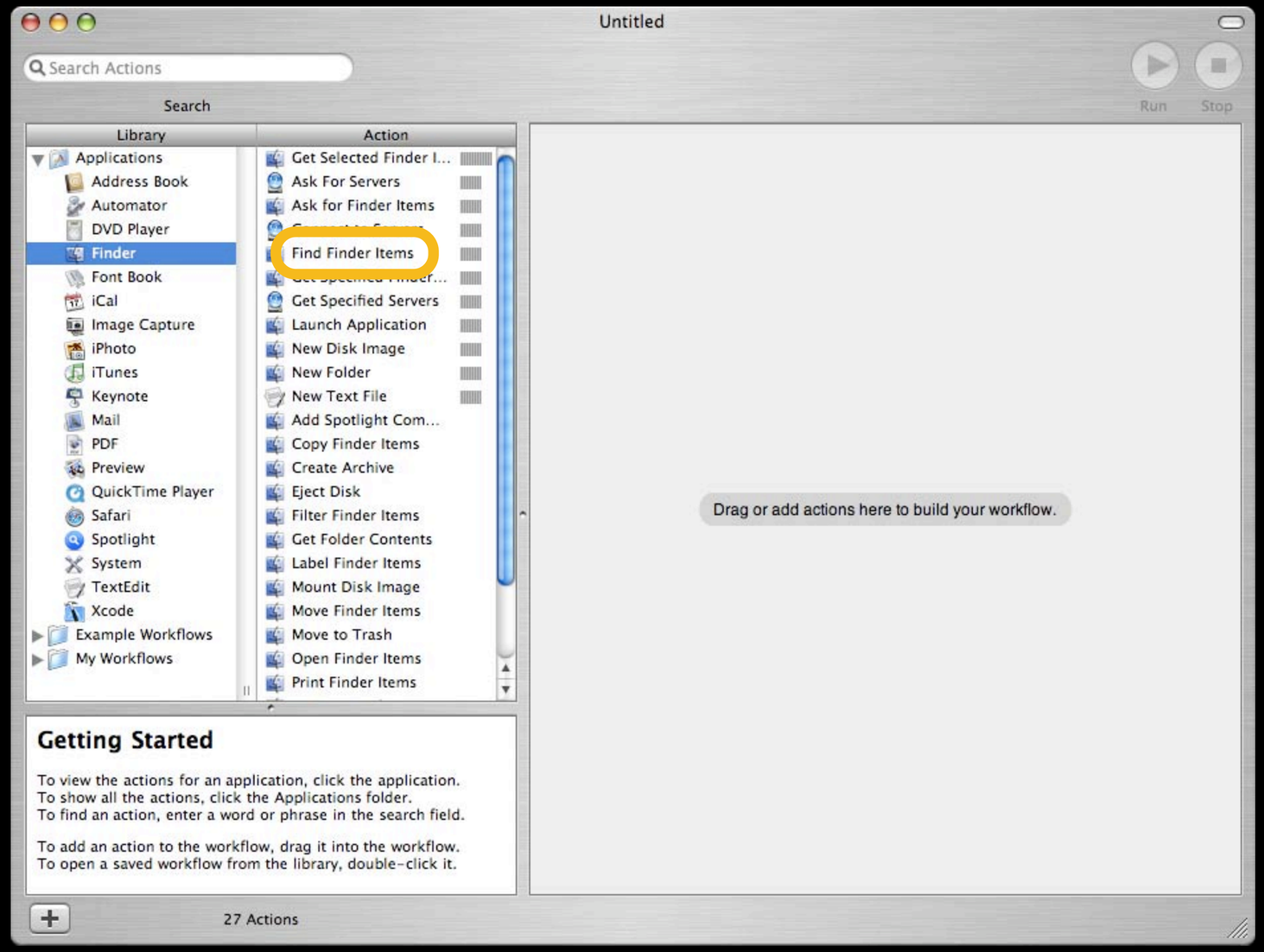
- AMApplication
- AMCategory
  - Use one of ours
  - Create your own
- AMIconName
- AMName



# Info.plist Keys

## General

- AMApplication
- AMCategory
  - Use one of ours
  - Create your own
- AMIconName
- AMName



# Info.plist Keys

## AMAccepts

- Specifies what input is accepted
- Optional boolean
- Types array
  - Array elements are Uniform Type Identifier strings

# Info.plist Keys

## AMAccepts

- Specifies what input is accepted
- Optional boolean
- Types array
  - Array elements are Uniform Type Identifier strings

## AMProvides

- Specifies what output is provided
- Types array

# Info.plist Keys

## AMAccepts/AMProvides guidelines

- Focus on interoperability
- Make AMAccepts and AMProvides types arrays as specific as possible
- Specify multiple types if applicable
  - e.g. iPhoto photos and albums
- If your action doesn't need input, e.g., View Results
  - Return input as output
  - Leave types general (default)

# Info.plist Keys

## Conversion actions

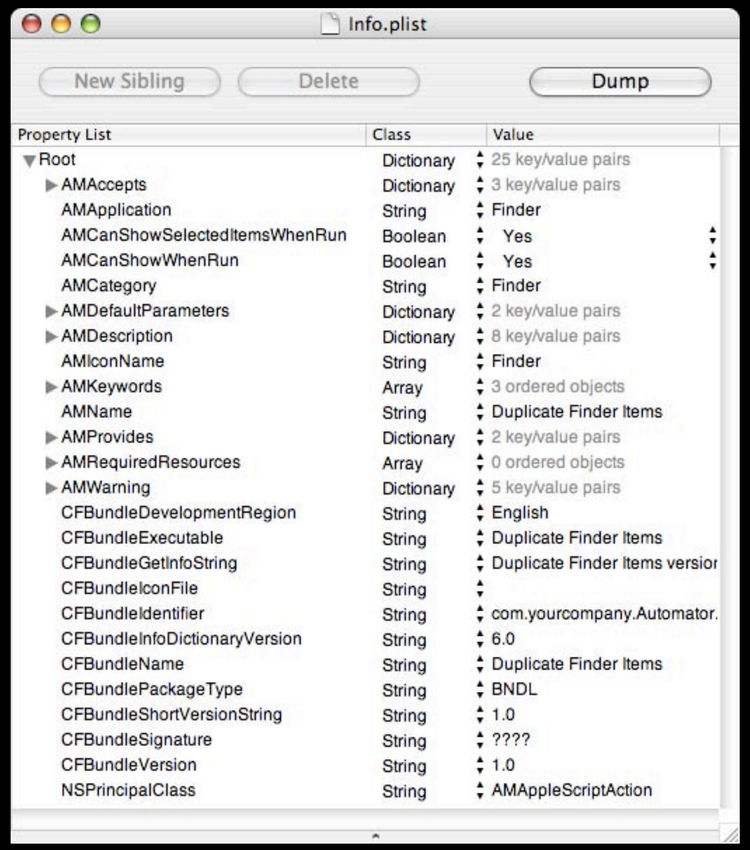
- Link actions whose types don’t match
  - Equivalent to AppleScript coercions
- Create for your new types
- Build with standard project template
- Differences from regular action
  - Bundle extension of .caction
  - No UI
  - Added automatically and invisibly (but appear in log)

| Hands On: Writing Cocoa Automator Actions | Session 137<br>Thursday 3:30PM |
| --- | --- |

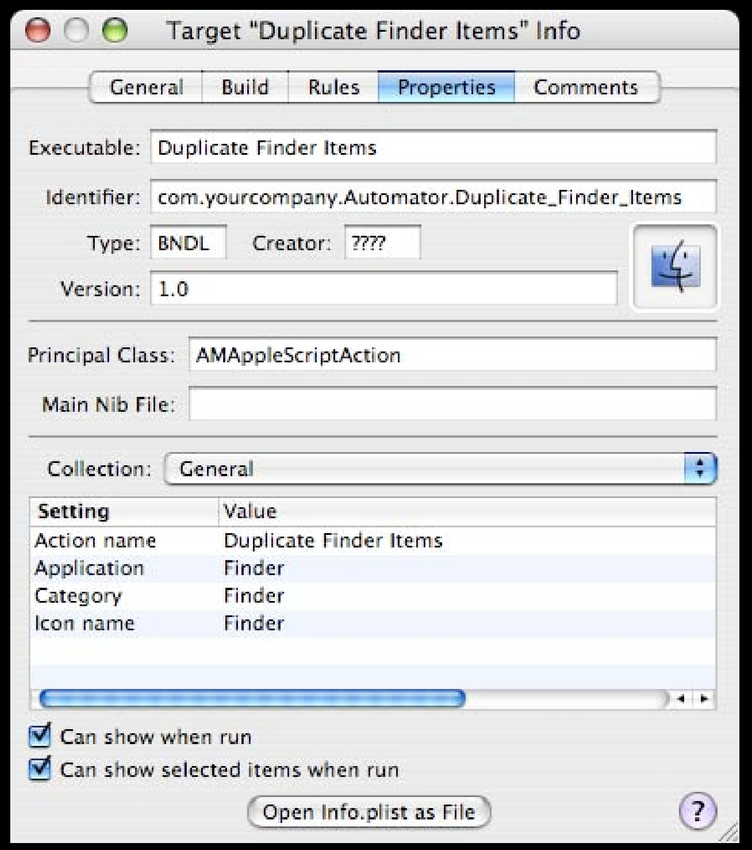
# What You Will Do in Step 2

- Edit the Info.plist file directly using Property List Editor
- Configure the action target inspector in Xcode

**2.0**




**2.1**



# Project Layout

## Step 2

- main.applescript
- Info.plist
- English.lproj
  - InfoPlist.strings
  - main.nib

# Hands On:

## Duplicate Finder Items Step 2

**Configure Info.plist**

http://developer.apple.com/wwdc2005/

# What You Learned in Step 2

- How to configure the action’s Info.plist file

# Duplicate Finder Items Step 3

**Add User Interface Elements Controlled by Cocoa Bindings**

http://developer.apple.com/wwdc2005/

# Creating an Action

## Design guidelines

- Help you achieve consistent look and feel
- Use our actions as examples
- Guidelines in the Automator Programming Guide
- Philosophy
  - Small
  - Discrete
  - Keep separate actions separate
    - ✅ Do let users combine separate actions in a workflow
    - ❌ Don’t provide a larger, more complex single action
  - Keep it simple

# Creating an Action

## AMDefaultParameters

- Default values for keys managed by Cocoa Bindings
- Use
  - Boolean for checkboxes
  - Integer for popup menu index
  - Integer, Real, or String for text fields
  - String for AMPathPopUpButton

# What You Will Do in Step 3

- Edit main.nib to build action's user interface
  - Use Directory Chooser AMPathPopUpButton
  - Follow relevant design guidelines
- Connect user interface elements to script using Cocoa Bindings
- Add AMDefaultParameters entries for Bindings
- Add AppleScript code to use Bindings

# Project Layout

## Step 3

- main.applescript
- Info.plist
- English.lproj
  - InfoPlist.strings
  - main.nib

# Hands On:

## Duplicate Finder Items Step 3

**Add User Interface Elements Controlled by Cocoa Bindings**

http://developer.apple.com/wwdc2005/

# What You Learned in Step 3

- How to build the action's user interface
  - Using Directory Chooser AMPathPopUpButton
  - Following relevant design guidelines
- How to connect user interface elements to the script using Cocoa Bindings

# Duplicate Finder Items Step 4

**Add User Interface Elements Controlled by AppleScript Studio**

http://developer.apple.com/wwdc2005/

# What You Will Do in Step 4

- Add behavior to action's user interface using AppleScript Studio
  - Attach and implement important action view handlers
    - awake from nib
    - opened
    - parameters updated
- Test action with Show Action When Run option

# Project Layout

## Step 4

- main.applescript
- ui.applescript
- Info.plist
- English.lproj
  - InfoPlist.strings
  - main.nib

# Hands On:

## Duplicate Finder Items Step 4

**Add User Interface Elements Controlled by AppleScript Studio**

http://developer.apple.com/wwdc2005/

# What You Learned in Step 4

- How to add behavior to action's user interface using AppleScript Studio
  - Attaching and implementing action view handlers
- How to ensure Show Action When Run option works correctly

# Duplicate Finder Items Step 5

**Make Localizable**

http://developer.apple.com/wwdc2005/

# What You Will Do in Step 5

- Make Info.plist entries localizable
- Add a Localizable.strings file
- Read strings from Localizable.strings file in scripts

# Project Layout

## Step 5

- main.applescript
- ui.applescript
- Info.plist
- English.lproj
  - InfoPlist.strings
  - Localizable.strings
  - main.nib

# Hands On:

## Duplicate Finder Items Step 5

**Make Localizable**

http://developer.apple.com/wwdc2005/

# What You Learned in Step 5

- How to make Info.plist entries localizable
- How to use localizable strings in scripts

# Duplicate Finder Items Step 6

**Test and Debug**

http://developer.apple.com/wwdc2005/

# Creating an Action

## Test and debug

- Actions
  - Build and Go
    - Uses Automator custom executable with launch argument

      –action “<project name>.action”
    - Runs amlint before launching Automator
  - Build and Debug
    - Step and view variables using the AppleScript debugger
- Workflows
  - Add View Results to see output of previous action
  - Add Ask for Confirmation to pause/cancel execution

# Creating an Action

## amlint

- Action verification tool
- New in Xcode 2.1
- Integrated into the action project templates
  - Run automatically during action build
- You can also run it on existing .action bundles

# What You Will Do in Step 6

- Correct nib errors using amlint output
- Debug action using
  - `log` statements
  - AppleScript debugger

# Hands On:

## Duplicate Finder Items Step 6

**Test and Debug**

http://developer.apple.com/wwdc2005/

# Creating an Action

## Installation locations

- Apple-provided actions
  - /System/Library/Automator
- Your actions
  - ~/Library/Automator
  - /Library/Automator
  - /Network/Library/Automator
  - YourApplication.app/Contents/Library/Automator

# What You've Learned

- How to build, test, debug, and install an action using
  - AppleScript
  - AppleScript Studio
  - Cocoa Bindings
  - Interface Builder
  - Xcode

# What Next?

- Experiment with this sample project
- Read the documentation
- Explore the example projects
  - /Developer/Examples/Automator

# What Next?

- Experiment with this sample project
- Read the documentation
- Explore the example projects
  - /Developer/Examples/Automator
- Write and ship lots of great actions

# More Information

http://developer.apple.com/wwdc2005/

# Documentation, Sample Code, and Other Resources

http://developer.apple.com/wwdc2005/

# Related Sessions

| | |
|---|---|
| Getting the Most Out of Cocoa Bindings | Session 136<br>Thursday 2:00PM |
| Hands On: Writing Cocoa Automator Actions | Session 137<br>Thursday 3:30PM |
| Hands On: AppleScript Studio in Depth | Session 406<br>Friday 9:00AM |
| Lab: Automator Action Development | Application Tech Lab<br>Friday 10:30AM |
| Lab: AppleScript Studio | Application Tech Lab<br>Friday 12:00PM |
| Building Automator Actions for System Administrators | Session 645<br>Friday 5:00PM |

# Who to Contact


**Sal Soghoian**
AppleScript and Automator Product Manager
sal@apple.com

**Todd Fernandez**
AppleScript and Automator Engineering Manager
trf@apple.com


**Automator Developers**
Mailing List
automator-dev@lists.apple.com

# Q&A

**John Montbriand**
DTS Engineer

http://developer.apple.com/wwdc2005/

# WWDC2005

Worldwide Developers Conference